REPORT DOCUMENTATION PAGE

1. REPORT DATE *10-22-2003*	2. REPORT TYPE FINAL	3. DATES COVERED *(From - To)* 08/1999 — 07/2002

4. TITLE AND SUBTITLE

Software, Programming, and Run-Time Coordination for Distributed Robotics

5a. CONTRACT NUMBER DABT63-99-1-0022

5b. GRANT NUMBER DABT63-99-1-0022

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Roderic Grupen

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Computer Science Department
University of Massachusetts
140 Governors Drive
Amherst, MA 01003

8. PERFORMING ORGANIZATION REPORT NUMBER
01

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington, VA 22203-1714

10. SPONSOR/MONITOR'S ACRONYM(S)
DARPA

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release. Distribution is unlimited.

20031117 051

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Our project created and disseminated new technologies for coordinating the behavior of large numbers of form factor constrained mobile robots. Simulations and robotic experiments were created to demonstrate coordination constraints expressed in a controlled theoretic framework in a manner that guarantees performance (time, energy, and comm bandwidth), scales to hundreds of individuals, and supports optimization via reinforcement learning to acquire distributed control policies.

We have implemented 10 "uBots" to realize our SDR concept on which we have demonstrated adaptive impedance control, multi-robot coordination and dynamic role assignment, a real-time process scheduler, and wearable interfaces. Applications, including: search and mapping; leader-follower control; and multi-robot behavior for preserving network connectivity among coordinated peers. Performance bounds on an n-robot teams using network-distributed interfaces have been demonstrated as well.

Twelve students are involved in our SDR project. We ported control code for autonomous grasping and manipulation gaits for robot hands to our colleagues at NASA-JSC for use in the Robonaut program and our simulation for multi-robot search controllers that maintain line of sight was transferred to colleagues at SPAWAR.

15. SUBJECT TERMS

redundant distribute systems, learning, network connectivity

16. SECURITY CLASSIFICATION OF:

a. REPORT	b. ABSTRACT	c. THIS PAGE
Unclassified	Unclassified	Unclassified

17. LIMITATION OF ABSTRACT
UL

18. NUMBER OF PAGES
8

19a. NAME OF RESPONSIBLE PERSON
Roderic Grupen

19b. TELEPHONE NUMBER *(include area code)*
(413)545-3280

Objective: The goal of our contract is to produce and deliver technologies for coordinating the behavior of large numbers of form factor constrained mobile robots. These systems are treated as massively redundant and therefore present many solutions for a given problem. The project contributes technologies that extend conventional approaches for planning coordinated behavior and for programming reactive swarms. The approach is informed by biology, where many examples of societal organizations (specifically in insects) exist, and by control theory, where methods exist to coordinate and guarantee performance of redundant systems. The project has realized both simulated and real experimental work and has demonstrated coordination constraints expressed in a control theoretic framework. The project will demonstrate how a distributed implementation with localized computation and communication can guarantee performance, can scale to hundreds or thousands of individuals, and how optimization techniques (like reinforcement learning) can be used in this context to acquire skillful, distributed control policies.

Approach: The UMass approach brings traditions in control, scheduling, real-time computing, and discrete event systems together with dynamic programming-based reinforcement learning. The UMass project advances a programming and analysis methodology for distributed robots with which to build Behavior Programs (B-Pgms) for coordinating the activity of multiple mobile robots with stringent form factors. Our methodology involves both simulated and real experimental work. We wish to demonstrate how coordination constraints can be expressed in a control theoretic framework, how global specifications can be guaranteed in a distributed implementation with localized computation and communication, and how this approach scales to hundreds of individuals. We also wish to demonstrate that exploration-based learning algorithms (like reinforcement learning) can be used in this context to acquire skillful, distributed control policies. The control framework employs a combination of closed-loop controllers that actively preserve local properties in the system, a method for combining them to address multiple, concurrent objectives using cascaded null space projections, and coordination templates for combining controllers across platforms that actively maintain constraints for pairs of robots. Pairwise policies have been used to demonstrate search-and-map, attack-repel, leader-follower, and network connectivity behavior among groups of independent robots.

The off-line programming environment uses of a closed-loop sensorimotor instruction set characterized as a Markov Decision Problem (MDP) and constrained in a Discrete Event framework to permit only correct programs in the distributed system. Correctness includes notions of safety, real-time and memory constraints, kinematic performance limitations, and functional specifications. Both user and automated programming techniques are employed to address currently inadequate levels of programmability in distributed, sensor-based applications.

A run-time environment has been constructed that manages the execution of comprehensive, distributed software on modest target processors. The goal is to accumulate a library of correct programs cached in a large flash memory to respond to a variety of tasks and

contingencies using code that cannot reside simultaneously in working memory. Behavior programs are replicated in memory on each robot to coordinate distributed behavior that is coarsely synchronized by low-bandwidth interprocess communications. Global correctness specifications are thus preserved in the compute-bound, distributed controller.

SDR Concept and Platform

We have constructed 10 prototype platforms based on the StrongARM processor. Examples of the control methodology have been developed whereby simple skill sets in single robots are acquired and used to express peer-to-peer coordination strategies. Finally, team oriented collaborations have been constructed from these peer-to-peer policies.

uBots interact in the context of an information framework that is serviced by other independent teams of uBots and human operators. We are trying to make optimal control decisions hierarchically by gathering "intelligence" in the form of additional state from robot teams with narrow, focused objectives. Teams of teams are configured and arranged to meet control objectives that are robust with respect to environmental variation and respond to run-time feedback. We have implemented such a hardware platform in the form of multi-uBots and auxiliary sensors that are distributed spatially and networked. The sensor network consists of panoramic cameras, pan/tilt/zoom (PTZ) cameras, pyroelectric sensors, and a communication network over which to combine features across multiple sensors.

Virtual and augmented reality interfaces have been constructed to exchange information between humans and a distributed set of robots. The wearable form factor enables use of the interface in the field, where the user becomes a participant in the data gathering process. The augmentation of the external world enables a user to observe otherwise invisible world states (e.g. chemical hazards), and observe areas of the local space that are occluded from the user's current point of view. This information can be painted on top of the external world (augmenting the user's sensory abilities), or can virtually bring the user to a different location from which to observe the world.

Our project aimes to understand how exploit structure for learning distributed robot behavior. We have developed various incarnations of an interaction-based state estimator to structure the representation of multi-objective actions. We are studying learning algorithms for reconfigurable stereo vision and view planning for localizing and tracking human subjects.

Systems and Algorithms

We have made steady progress in algorithm and system developments in the following aspects.

Adaptive Impedance Control

We have continued to develop an interaction-based state estimator for optimizing tracking behavior of navigation controllers. A new representation is being used called impedance

codes (or i-codes) that observes tracking performance through multiple impedance filters and employs one of them to actuate the motor system. We have constructed switching policies indexed by i-codes to optimize performance with respect to time, energy, and precision. Moreover, we are now experimenting with concurrent combinations of time-, energy-, and precision-optimal policies that automatically encode multiple objectives in a single control expression.

Multi-Robot Coordination and Roles

We conducted research regarding adaptive role switching among robots to increase overall performance of the team. We examined the task of exploring a set of rooms while maintaining line-of-sight connectivity among a team of robots. We have developed a model of interaction-based state that classifies the dynamics of the active controllers of the constituent robots. We anticipated that learning to switch roles on this basis would increase the efficiency of the search while maintaining the line of sight constraint. However, our experiments indicate that there is, in general, little advantage in this task to switching role.

Real-Time Process Scheduling

Intelligent coordination schemes form new peer-to-peer relationships at run-time to deal opportunistically with features of the environment that are discovered during execution. To do so, we have begun to develop a scheduling algorithm that can verify the schedulability of a variety of expected run-time configurations. Given a distributed robotic task set in which sensor and actuator tasks have temporal and locality constraints and periods, a feasible schedule is derived that satisfies all constraints and guarantees the deadlines of each task instance. Prior analysis of control configurations will enable the uBot team to consider distributing computational load the same way that it considered distribting motor and sensory tasks.

The scheduling algorithm has been implemented and the effect of the number of processors, the task size, and communication costs have been tested by the simulation. We are using the algorithm to do a detailed run-time analysis for our leader-follower control configuration. Results provide an upper bound on the number of robots that may be bound together in a coordinated behavior before control processes are no longer schdulable. At this point, team hierarchies must be employed to coordinate groups of robots.

Wearable Interfaces

Our implementation is based on a Xybernaut wearable computer as a computational and display device. A three-dimensional graphics engine renders virtual objects on the heads-up-display at approximately 10 frames-per-second. Head orientation is sensed in real time using an accelerometer-corrected gyroscope. Current head position is provided either by our smart sensor network or via GPS. Demonstrations have been constructed in which a human collaborator subscribes to the sensor stream generated by a distributed robot team to:

- see through visual occluders;

- annotate the environment with auxiliary cues highlighting hazards, safe paths, and exits in a search-and-rescue scenario; and

- illustrate the performance and progress of a distributed robot search task

The VR interface allows the user to walk around within this virtual space and even achieve birds-eye views of the state of the environment.

Accomplishments: We have produced deliverable technologies in several of the research foci.

1. A parametric motion controller that has been used to implement interesting classes of coordinated behavior, including:

   - search and mapping behavior,
   - leader-follower behavior,
   - a multi-robot behavior for preserving network connectivity among peers in a co-ordinated task, and
   - a bounded-overwatch behavior that localizes individuals within a coordinated team using virtual stereo

2. Demonstrated control expressions that compile into distributed implementations that satisfy global specifications and demonstrated how equivalent functional configurations vary with respect to time, energy, and comm bandwidth,

3. Mechanisms for recovering hidden state in distributed systems.

4. 10 UMass "uBots" each with onboard StrongArm processors, embedded Scenix controllers for IR obstacle detect, inter-robot communication, and RF data link. Monocular color vision systems have been prototyped and installed on some of the robots to transmits video to an off-board a vision host (a 933MHz Pentium III).

5. Policies for Group Coordination: line-of-sight constraints in communicating teams of robots, and bounded overwatch strategies for preserving a global frame.

6. Policies for exploration and map making while maintaining localization

7. Demonstrated RT analysis and performance bounds on a n-robot search configuration.

8. Virtual environments for robot control and situation assessment on the basis of percepts generated during a distributed multi-robot search task at 10 frames per second.

9. A network-distributed object representation system that distributes three-dimensional, virtual objects across a set of servers located on a common network.

   (a) Visualization of the UMass Torso robot is reflected in real time into a virtual environment.

   (b) "X-ray vision" demonstration. The user is able to perceive remote, moving objects that are occluded by walls or other objects.

   (c) Dynamic display of distributed robot search task: robot positions are displayed in a virtual world; as the robots discover walls and target objects in this space, these objects are added to the model. The VR interface allows the user to walk around within this virtual space and even achieve birds-eye views of the state of the environment.

   (d) Mosaicing visual streams into a texture-mapped virtual environment.

10. UMass has begun an implementation of a grasping control architecture for the NASA Robonaut program based on the ideas developed under the SDR and MARS programs.

The following text discusses our progress in more detail and on several dimensions of the project.

Hardware - UMass UBots The UMass SDR project has produced 10 motion platforms (UMASS uBots) that comply to the form factor guidelines underlying the program. Coordinated controllers have been implemented on the two existing platforms. The UMASS uBot consists of multiple layers: a vision (and future acoustic) sensor layer, an IR layer for obstacle detection, future capacity for an IRDA broadcast communication channel, a DIMMPC implementation with RT kernel, and a QNX StrongArm version, and I2C communications between embedded Scenix subsystems. Currently, IR and motor power subsystems are running with stable firmware. In addition, an RF rangeLAN point-to-point communication facility for robots to communicate with their peers has been implemented. The first robot demonstrations of the control structures proposed are running in the laboratory and demonstrations are being constructed for a series of multi-robot search tasks in a model office-style interior space.

Software - Control Polices and Representations Intra-robot communications between multiple embedded controllers is complete and tested. This firmware is the subject of the first application of the RT analysis software. Among the first results, several distributed leader-follower control configurations have been guaranteed for a bounded number of robots (up to 35 for one configuration). RF communications between robots through a logical global memory model is also complete and currently under testing. Applications software has also been developed for a variety of control tasks including: multi-robot search tasks, bounded overwatch, leader-follower applications, and swarm controllers. In addition, laboratory benchtop vision systems for our platforms have been completed and data sets have been generated with which to learn to recognize

peers in a partially occluded environment and to extract features of the environment as constraints to motion control tasks.

Developmental Programming Paradigm Examples of the control methodology whereby simple skill sets in single robots precede peer-to-peer coordination strategies, and finally, team-oriented collaborations have been constructed. The idea is to compile control knowledge incrementally in a developmental sequence.

Experiments with the interaction-based representation for motor control have been demonstrated in simulation that estimate control state by observing the transient response of a discrete set of linear impedance controllers. Results verify the generalization of policies based on these representations to novel run-time contexts. Policies have been constructed for optimal energy, time, and precision controllers.

To structure peer-to-peer interactions, a control constraint between coordinated pairs of robots has been developed called the "push" and "pull" kinematic conditions. These conditions specify kinematic properties of a chain of interacting robots that can express leader-follower relationships. Moreover, these kinematic conditions have been used to express line-of-sight connectivity in a communication network with an arbitrarily large number of robots.

Stable Real-Time Adaptation An extension to vanilla C-syntax has been developed with which to support RT analysis in concurrent, embedded applications. Programs written with *.be* code extensions contain descriptions of periodic the control tasks that specify the robots behavior under real-time constraints. Design rules are employed to conquer the complexity of RT analysis using "divide and conquer" techniques, by folding together *.be* program phases that have identical time/resource requirements, and by partitioning *.be* programs into adaptive sets. We have implemented a real-time kernel and a graphical user interface for developing *.be* programs for the DIMMPC. The GUI serves as the front end for advanced off-line analysis and compilation of the *.be* programs. Once the RT programs have been specified and compiled, the system guarantees that they will meet runtime deadlines. We plan to conduct a complete analysis and verification of the platform firmware during summer of 2001 and subsequently, to verify higher-level control applications as well.

Technology Transition: Twelve students are involved in our SDR project.

We are continuing to identify opportunities for the direct transition of technology under the project to DOD applications. We recently acquired a small NASA contract to deliver our controllers for applications to the Robonaut program.

We recently ported control code for autonomous grasping and manipulation gaits for robot hands to our colleagues at NASA-JSC for use in the Robonaut program and our simulation for multi-robot search controllers that maintain line of sight was transferred to colleagues at SPAWAR.

Publications:

1. Uppala, S., Karrupiah, D., Brewer, M., Ravela, S. and Grupen, R., "On Viewpoint Control," 2002 International Conference on Robotics and Automation, ICRA02, Washington, DC, May, 2002.

2. Holness, Karuppiah, Uppala, Grupen, Ravela, "A Service Paradigm for Reconfigurable Agents, " 2nd International Workshop on Infastructure, Scale and Scaleable MAS Montreal, Quebec, Canada. May 2001 - 5th Autonomous Agents conference in Montreal

3. Piater, Grupen, "Feature Learning for Recognition with Bayesian Networks," 15th International Conference on Pattern Recognition, Barcelona, Spain, August, 2000.

4. Sweeney, J., Brunette, TJ, Yang, Y., and Grupen, R., "Coordinated Teams of Reactive Mobile Platforms," in the Proceedings of the 2002 IEEE Conference on Robotics and Automation, Washington, D.C. May, 2002.

5. Rosenstein, M., Grupen, R., "Velocity-Dependent Dynamic Manipulability," in the Proceedings of 2002 International Conference on Robotics and Automation, Washington, D.C. May, 2002

6. Popplestone, Grupen, "Symmetries in World Geometry and Adaptive System Behaviour," 2nd International Workshop on Algebraic Frames for the Perception-Action Cycle (AFPAC 2000), September 10-11, 2000, Kiel, Germany.

7. Karuppiah, Zhu, Shenoy, Riseman, "A fault-tolerant distributed vision system architecture for object tracking in a smart room," IEEE Second International Workshop on Computer Vision Systems, Vancouver, Canada, July 7-8, 2001

8. Piater "Visual Feature Learning," Ph.D. Dissertation, February 2001.

9. Zhu, Rajasekar, Riseman, Hanson. "Panoramic virtual stereo vision of cooperative mobile robots for localizing 3D moving objects," Proceedings of IEEE Workshop on Omnidirectional Vision - OMNIVIS'00, Hilton Head Island, 29-36.

10. Zhu, "Omnidirectional stereo vision," Workshop on Omnidirectional Vision Applied to Robotic Orientation and Nondestructive Testing (NDT), The 10th IEEE International Conference on Advanced Robotics, August 22-25, 2001, Budapest, Hungary.

11. Zhu, Karuppiah, Riseman, Hanson. "Panoramic virtual stereo vision in cooperative mobile robots for localizing moving objects," submitted to Computer Vision and Image Understanding, 2001.